DESCRIPTION

PROCESSOR, MULTIPROCESSOR SYSTEM, PROCESSOR SYSTEM, INFORMATION PROCESSING APPARATUS AND TEMPERATURE CONTROL METHOD

TECHNICAL FIELD

The present invention relates to a processor technology and, more particularly, to a processor, a multiprocessor system, a processor system, an information processing apparatus and a temperature control method capable of controlling heat value.

BACKGROUND TECHNOLOGY

As the process of fabrication becomes increasingly finer and components are more highly integrated, heat value has become an important concern in designing an LSI as a parameter indicating a limit of chip performance. As a chip is heated to a high temperature, the chip may operate improperly or the long-term reliability thereof decreases. Therefore, various countermeasures for heat release are taken. For example, a radiator fin may be provided on top of the chip to allow heat from the chip to escape.

Since the distribution of power consumption on a chip is not uniform, the problem of "hot spot", a part of the chip at an abnormally high temperature, cannot be avoided.

Accordingly, a study has been made of processor task scheduling based on the distribution of power consumption on a chip (see, for example, patent document No. 1).

[patent document No. 1]

United States published patent application 2002/0065049

When a part of a chip release heat, a high-temperature area extends with time into an area surrounding the heated part by heat conduction until the temperature of the whole chip is increased eventually. A related-art countermeasure for heat release is to observe the temperature distribution of the whole chip on a macroscopic scale and radiates heat over time (for example, from several seconds to one minute). As such, time response is poor. Some highly-integrated LSIs fabricated recently consume power on the order of 10-100 KW per chip. Unless heat is radiated in a time span on the order of 10-100 microseconds, mal-operation may be caused due to sudden rise in temperature. In a case where heat is released locally but the chip as a whole is at a safe temperature, lowering the operating frequency of the chip is inefficient in that it results in sacrificing the processing performance of the whole processor. In this background, we have come to be aware of the need to spot an area of heat release when a part of a chip releases heat before a high-temperature area grows by heat conduction and to exercise a countermeasure on a microscopic level for preventing further temperature rise.

DISCLOSURE OF THE INVENTION

The present invention has been made in view of the aforementioned problems and its object is to provide a processor, a multiprocessor system, a processor system, an information processing apparatus and a temperature control method capable of controlling the heat release of a chip on a very fine scale both in terms of time and space.

The present invention according to one aspect provides a processor. The processor according to this aspect comprises: a heat release frequency holding unit which holds a heat release frequency of a plurality of blocks subject to heat release control; a heat release identifying unit which identifies a block involved in the execution of each execution unit comprising at least one instruction, and which identifies a heat release coefficient related to a heat value of the identified block; and a heat release frequency adder unit which cumulatively increases, for each execution unit, the heat release frequency of the identified block by referring to the heat release coefficient, as the execution of instructions proceeds.

A block is a minimum unit produced by segmenting the processor area. The processor area is segmented into blocks each characterized by a localized peak in heat release. For example, the block may be a discrete component constituting the processor such as a transistor. Alternatively, the block

may be a set of some number of components. In the case of multiprocessor that includes a plurality of processors, the block may be a segment in each of the individual processors. Alternatively, the whole processor may represent a block.

An execution unit may be an instruction code comprising each step. Alternatively, an execution unit may be a subroutine or a task constituted by a collection of instruction steps.

The size of a block and the granularity of an execution unit can be flexibly designed depending on the required precision in heat release control and the condition imposed on processor design.

The present invention according to another aspect also provides a processor. The processor according to this aspect comprises: a heat release frequency holding unit which holds a heat release frequency of a plurality of blocks subject to heat release control; a heat release frequency adder unit which cumulatively increases, for each execution unit comprising at least one instruction, the heat release frequency of a block involved in the execution of the execution unit by referring to a heat value estimated for the execution unit, as the execution of instructions proceeds; and a scheduler which schedules instructions to be executed in accordance with the heat release frequency of the blocks.

The present invention according to still another aspect also provides a processor. The processor according to this

aspect comprises: a heat release frequency holding register which holds a heat release frequency of a plurality of blocks subject to heat release control; a heat release coefficient profile which stores, in correspondence with each other, an instruction to be executed, a block involved in the execution of the instruction and a heat release coefficient related to a heat value of the involved block; a decoder which analyzes an instruction to be executed so as to identify, for each execution unit comprising at least one instruction, the block involved in the execution of the instruction and the associated heat release coefficient, and which stores identified information in the heat release coefficient profile; a heat release frequency adder unit which cumulatively increases, for each execution unit, the heat release frequency of the identified block by referring to the heat release coefficient profile, as the execution of instructions proceeds; and a scheduler which schedules instructions to be executed in accordance with the heat release frequency of the blocks.

The present invention according to yet another aspect provides a multiprocessor system. The multiprocessor system includes a plurality of subprocessors and a main processor. The main processor comprises: a heat release frequency holding register which holds a heat release frequency of a plurality of blocks in each of the subprocessors; a heat release identifying unit which identifies a block involved in

the execution of each execution unit comprising at least one instruction, and which identifies a heat release coefficient related to a heat value of the identified block; a heat release frequency adder unit which cumulatively increases, for each execution unit, the heat release frequency of the identified block by referring to the heat release coefficient, as the execution of instructions proceeds; and a scheduler which allocates instructions to be executed among the plurality of subprocessors in accordance with the heat release frequency of the blocks.

The present invention according to another aspect also provides a processor. In this processor, a decoder for decoding an instruction to be executed is provided with the function of analyzing heat release of a block in the processor involved in the execution of the instruction.

The present invention according to still another aspect provides a temperature control method. In this temperature control method, heat release occurring as instruction codes are executed is detected in units of blocks, by cumulatively increasing, for each execution unit comprising at least one instruction, a heat release frequency of a block involved in the execution of the execution unit, as the execution of instruction codes proceeds, and by holding the accumulated heat release frequency in a register.

The present invention according to yet another aspect also provides a temperature control method. This temperature

control method cumulatively increases, for each execution unit comprising at least one instruction, a heat release frequency of a block involved in the execution of the execution unit by referring to a heat value estimated for the execution unit, as the execution of instructions proceeds, holds the accumulated heat release frequency in a register, and schedules instructions to be executed in accordance with the heat release frequency of blocks held in the register.

The present invention according to another aspect also provides a temperature control method. This temperature control method estimates, for each execution unit comprising at least one instruction, a heat value of blocks in each of processors in a multiprocessor system, predicts temperature variation in the blocks as the execution of instructions proceeds, and allocates instructions to the processors in accordance with the predicted temperature variation.

Arbitrary combinations of the aforementioned constituting elements, and implementations of the invention in the form of methods, apparatuses, systems and computer programs may also be practiced as additional modes of the present invention.

According to the present invention, heat value generated from the processor is controlled and mal-operation is prevented, without incurring degradation in processor performance.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 illustrates the structure of a processor system according to a first embodiment of the present invention;

Fig. 2 is a detailed illustration of functional blocks in the processor system of Fig. 1 involved in heat release control;

Fig. 3 illustrates a heat release coefficient profile;

Fig. 4 illustrates a heat release frequency stored in an operational block heat release frequency register of Fig. 1;

Fig. 5 is a flowchart illustrating a procedure for heat release control in the processor system of Fig. 1; and

Fig. 6 illustrates the structure of a processor system according to a second embodiment of the present invention.

BEST MODE OF CARRYING OUT THE INVENTION

First embodiment

Fig. 1 illustrates the structure of a processor system according to a first embodiment of the present invention. The processor system includes a CPU core 100 and a main memory 110 which are connected to an address bus 28 and a data bus 30. The CPU core 100 designates an address in the main memory 110 and reads and writes data in the main memory 110. The CPU core 100 includes an instruction cache 12, an instruction decoder 14, an instruction scheduler 16, an execution unit 18, a heat release coefficient profile 20 and

an operational block heat release coefficient register 22. The main memory 110 stores an instruction 24 and an operational result 26.

The instruction 24 read by the CPU core 100 from the main memory 110 is cached in the instruction cache 12. The instruction decoder 14 sequentially decodes the instruction 24 cached in the instruction cache 12 and supplies the decoded instruction to the instruction scheduler 16. The instruction scheduler 16 schedules the instructions 24 by rearranging the order of execution of the instructions 24 and/or adjusting execution timing in accordance with the dependence on data of each of the instructions 24 decoded by the instruction decoder 14. The execution unit 18 executes the instruction 24 thus scheduled and writes the operational result 26 in the main memory 110.

The whole area occupied by the chip of the processor system chip is segmented into small areas subject to heat release control. Each of the small areas is referred to as a block or an operational block. The operational block may be a discrete transistor constituting the chip or a set of some number of transistors constituting the chip. The chip area is segmented into operational blocks each characterized by a localized peak in heat release. The size of the operational block may be determined as desired so as to meet a target precision in heat release control and specifications required of the processor. The chip area may be segmented into

operational blocks of uniform size or segmented according to the borders between the operational units into non-uniform sizes.

The heat release coefficient profile 20 is a stored profile which stores, in correspondence with each other, an instruction step, an operational block of the processor system involved in the execution of the instruction step and a heat release coefficient indicating the heat value of the operational block involved. An operational block heat release frequency register 22 is a register for holding an accumulated total of the heat release frequency of the operational blocks.

Fig. 2 illustrates functional blocks of the CPU 100 relative to heat release control. The instruction decoder 14 is provided with a function of analyzing heat release. The instruction decoder 14 identifies an operational block involved in the execution of each instruction step by referring to hardware information of the processor system, predicts heat value due to the operation of the identified operational block and determines a heat release coefficient. The instruction decoder 14 stores, in correspondence with each other, an instruction step, information on the location of the identified operational block and the associated heat release coefficient, in the heat release coefficient profile 20.

Fig. 3 illustrates an example of the heat release

coefficient profile 20. An instruction step 40, an operational block location 42 and a heat release coefficient 44 are stored in the profile in correspondence with each other. The instruction step 40 represents a step-by-step instruction decoded by the instruction decoder 14 and is identified by a command such as MOV (transfer), ADD (addition), LD (load) and ST (store), followed by an argument. For example, the instruction step "MOV AX BX" is an instruction dictating the substitution of the content of an arithmetic register BS of the CPU core 100 into an arithmetic register AX. The instruction step "LD AX 2D" is an instruction dictating the load of data at an address 2D of the main memory 110 into the arithmetic register AX of the CPU 100.

The operational block location 42 is given as an index of matrix occurring when a die on the processor system package is segmented into rectangular areas. The heat release coefficient 44 is a numerical value determined from predicted value of heat generated when the operational block located in an area designated by the operational block location 42 executes the instruction.

By using results collected when the execution of program codes is simulated in a logical simulation conducted upon completion of the LSI layout of the processor system, variation in the on-off condition of the processor transistors can be traced at small time intervals. Using

these results allows complete analysis capable of identifying an operational block that goes active in each instruction step.

The heat value of the operational block is predicted by using a CAD tool or the like when designing a circuit in consideration of the static temperature characteristic of the operational block. The static temperature characteristic of the operational block is primarily quantified by referring to the physical characteristics or the relative location of the components. For example, a p-channel transistor and an n-channel transistor of a CMOS are simultaneously turned on when inversion occurs, producing a through current. The through current represents the majority of power consumption of a CMOS. The power consumption increases in proportion to the operating frequency of the CMOS. By estimating the power consumption, the heat value of the operational block can be predicted. The heat release coefficient is a value obtained by quantizing the predicted heat value. The information such as the results of simulation performed when designing the circuit or the predicted heat value can be referred to from the instruction decoder 14 as hardware information.

In the example illustrated in Fig. 3, the execution of the instruction step "MOV AX BX" involves operational blocks located at (2, 3) and (2, 4). The heat release coefficient is 2. The execution of the instruction step "LD AX 2D" requires an operational block at (2, 2) for its execution.

The heat coefficient is 1.

Referring back to Fig. 2, the instruction scheduler 16 schedules the instructions 24 decoded by the instruction decoder 14 in accordance with the dependence of each of the instructions on data and selects an instruction step to be executed next. The instruction scheduler 16 refers to the heat release coefficient profile 20 so as to identify the location and the heat release coefficient of the operational block involved in the execution of the selected instruction step and provides the identified information to a heat release frequency adder 32. The heat release frequency adder 32 reads the current heat release frequency of the operational block thus identified from the operational block heat release frequency register 22. The heat release frequency adder 32 adds the heat release coefficient supplied from the instruction scheduler 16 to the heat release frequency thus read and writes the sum to the operational block heat release frequency register 22.

The heat release frequency subtractor 34 reads the heat release frequency of the operational blocks from the operational block heat release frequency register 22 and performs a subtraction process based on a predetermined heat discharge constant. The heat release frequency subtractor 34 writes the result of subtraction in the operational block heat release frequency register 22. The heat release frequency subtractor 34 operates at a predetermined clock and

subtracts from the heat release frequency of the operational blocks in the operational block heat release frequency block register 22. With this, heat discharge occurring with time is reflected in the heat release frequency in the operational block heat release frequency register 22.

The heat release frequency subtractor 34 subtracts until the heat release frequency is zero. The larger the heat release frequency of the operational block, the larger the amount of subtraction. This is because the temperature is considered to drop faster due to heat discharge, as a difference between the temperature predicted from the heat release frequency of the operational block and the surrounding temperature grows. A measured value, an estimated value or a preset value may be used as the surrounding temperature.

Fig. 4 illustrates an example of heat release frequency stored in the operational block heat release frequency register 22. The heat release frequency of the operational blocks stored in the operational block heat release frequency register 22 is cumulatively increased by the heat release frequency adder 32 as the execution of instructions proceeds. The heat release frequency subtractor 34 subtracts from the accumulated frequency as time elapses.

The heat release frequency adder 32 and the heat release frequency subtractor 34 may adjust the heat release frequency of the operational blocks in consideration of the

dynamic aspect of the temperature characteristic of the operational blocks. The dynamic temperature characteristic of the operational blocks primarily depends on the history of execution of tasks or the condition of loads. In some cases, the temperature characteristic is affected by physical conditions. For example, more heat is likely to be generated when a task is simultaneously allocated to adjacent blocks than when the task is distributed among remote blocks. The heat release frequency adder 32 may add the heat release frequency by taking into account the condition of allocation of tasks to operational blocks or the interaction of heat released from adjacent operational blocks. The heat release frequency subtractor 34 may subtract from the heat release frequency in consideration of how heat discharge proceeds around the operational block.

Referring back to Fig. 2, the hot spot detector 36 detects an operational block with its heat release frequency, held in the operational block heat release frequency register 22, exceeding a threshold value. The hot spot detector 36 informs the instruction scheduler 16 of the location of the detected operational block. The threshold value is supplied as the heat release frequency corresponding to a temperature lower than a marginal temperature at which the operational block fails to operate properly. Accordingly, the hot spot detector 36 not only detects an operational block that actually turns into a hot spot but also detects an

operational block that is likely to turn into a hot spot in the future.

The instruction scheduler 16 reschedules the instructions by shifting the timing of execution by means of inserting a wait instruction before an instruction step involving for its execution the operational block determined to be a hot spot, or by giving priority to the execution of an instruction not involving for its execution the operational block determined to be a hot spot. As such, the instruction scheduler 16 performs scheduling that uses, as evaluating parameters, the quantified static/dynamic temperature characteristic of the operational blocks.

In the above-described structure, the instruction decoder 14 identifies operational blocks for each instruction step by referring to the hardware information and determines the associated heat release coefficient. Alternatively, the heat release coefficient may be embedded in the instruction as part of an operand in the process of generating the instruction code. For example, a programmer or a compiler may designate a heat release coefficient for each instruction.

Fig. 5 is a flowchart illustrating a procedure for heat release control executed by the instruction decoder 14 and the instruction scheduler 16 of the CPU core 100.

When an instruction to be decoded next is cached in the instruction cache 12 (Y of S10), the instruction decoder 14 decodes the cached instruction (S12). The instruction

decoder 14 identifies an operational block involved in the execution of the decoded instruction, determines the heat release coefficient of the involved operational block, and generates a heat release coefficient profile 20 (S14).

The instruction scheduler 16 schedules the instructions decoded by the instruction decoder 14 in accordance with the dependence of each of the instructions on data (S16). As the execution of the scheduled instruction codes proceeds, the heat release frequency adder 32 cumulatively adds the heat release coefficient to the heat release frequency of associated operational blocks held in the operational block heat release frequency register 22 (S18). In consideration of the interaction with adjacent operational blocks, the heat release frequency adder 32 may increase the amount of addition to the heat release frequency of the associated operational blocks when the heat release frequency of the adjacent operational blocks is large.

The hot spot detector 36 evaluates the heat release frequency of the operational blocks in the operational block heat release frequency register 22 so as to predict the temperature and determine whether there is any operational block that is likely to be a hot spot. When there is an operational block that is likely to be a hot spot (Y of S20), the instruction scheduler 16 reschedules the instructions by adjusting the order of execution and/or execution timing of the instructions such that the execution of an instruction

involving for its execution the operational block that is likely to be a hot spot is delayed (S22).

The heat release frequency subtractor 34 subtracts from the heat release frequency of the operational blocks in the operational block heat release frequency register 22 in accordance with a predicted value of heat discharge that occurs with time (S24). Control is then returned to S10, whereupon a series of steps is repeated until there are no more instructions to be decoded (N of S10).

According to the processor system of this embodiment, the heat value is predicted block by block and step by step as the execution of instruction codes proceeds. Accordingly, the temperature distribution of the processor system package as a whole is known accurately on a very fine scale both in terms of time and space. With this, even when intensive processing load is imposed on a specific operational block of the processor system such that a sudden temperature increase is predicted, the execution of instructions involving for their execution the specific operational block is scheduled on a real time basis, with the result that heat release is prevented from reaching its peak. Localized mal-operation due to heat release is thus prevented without sacrificing the processing performance of the processor as a whole.

Second embodiment

Fig. 6 illustrates the structure of a processor system according to a second example of the present invention. The

processor system according to the second embodiment is a multiprocessor system including two subprocessors 230a and 230b connected to a bus, in addition to a main processor 200 corresponding to the CPU core 100 of the first embodiment. The main processor 200 accesses a DRAM 220 via the bus and reads data therefrom. The main processor 200 caches the data in the cache 210. The main processor 200 allocates tasks to the two sub-processors 230a and 230b as appropriate for execution of a program.

The main processor 200 includes various functional blocks of the CPU core 100 described in the first embodiment including the instruction cache 12, the instruction decoder 14, the instruction scheduler 16, the instruction unit 18, the heat release coefficient profile 20, the operational block heat release frequency register 22, the heat release frequency adder 32, the heat release frequency subtractor 34 and the hot spot detector 36. The following description of these blocks concerns a difference in operation from the first embodiment.

In the processor system according to the second embodiment, the heat value of the processor system package as a whole, including the modules such as the main processor 200 and the subprocessors 230a and 230b, is a subject of heat release control. Segments in each of the modules such as the main processor 200 and the subprocessors 230a and 230b represent operational blocks described in the first

embodiment. Addition and subtraction of the heat release frequency for each operational block is performed as in the first embodiment.

The instruction scheduler 16 schedules the instructions by allocating instructions to the subprocessors 230a and 230b, in addition to changing the order of execution and adjusting the timing of execution as described in the first embodiment. Allocation of instructions by the instruction scheduler 16 to the subprocessors 230a and 230b may take place in units of instruction steps or in task units of certain sizes such as subroutines. The instruction scheduler 16 balances load by allocating instructions to the subprocessors 230a and 230b in accordance with the load condition. Further, the instruction scheduler 16 controls heat value by allocation adapted to the heat release condition of the operational blocks in the subprocessors 230a and 230b. For example, when an operational block that is likely to be a hot spot is detected in the subprocessor 230a, the instruction scheduler 16 allocates instructions to the subprocessor 230b.

The processor system according to this embodiment prevents a hot spot from being created in the processor system by averaging the temperature distribution in the whole processor system by allocating tasks to the subprocessors 230a and 230b in accordance with the heat release condition of the operational blocks in the subprocessors 230a and 230b.

Described above is a description based on the

embodiments. The embodiment is only illustrative in nature and it will be obvious to those skilled in the art that variations in constituting elements and processes are possible within the scope of the present invention.

Such variations will be described in the following section. In the described embodiments, a method as a countermeasure for heat release is described in which instructions are scheduled so that an operational block that is likely to be a hot spot is not heavily loaded. Alternatively, an operational block that is likely to be a hot spot may be cooled locally by a cooling nozzle. When heat release from an operational block cannot be controlled by instruction scheduling, an emergency measure may be taken. For example, the operating frequency of the whole processor may be reduced or the power supply voltage may be lowered.

In the described embodiments, heat value is estimated by running a simulator when designing a circuit and information on heat release coefficient is provided in the processor as hardware information. Alternatively, a sensor for measuring processor temperature may be provided so that heat release control according to scheduling described in the embodiments is performed by locally measuring the temperature of the operational blocks.

In the described embodiments, heat value is predicted for each instruction step and the heat release frequency is counted. Alternatively, heat value may be predicted in task

units of certain sizes such as subroutines so that the heat release frequency may be counted accordingly. Also, scheduling may be performed in task units instead of in units of instruction steps. There may be provided a switching control unit that switches between task-by-task scheduling and instruction-by-instruction scheduling. When the operational blocks of the processor are within a relatively safe temperature range, the heat release frequency counted in task units may be referred to for task-by task scheduling. When the temperature rises to a critical level, the heat release frequency of the operational blocks may be counted in units of instruction steps for fine-scale instruction-by-instruction scheduling.

In the described embodiments, heat release control is performed by the hardware of the processor system. Alternatively, an instruction code generator may be provided outside the processor system. In this case, instruction scheduling based on the analysis of instruction codes and the prediction of temperature may be performed as a software process outside the processor system so that the scheduled instructions are supplied to the processor.

There may be provided a checking unit which generates a heat release characteristic profile of each application, and which corrects profile information such as heat release coefficient by checking the profile against the temperature distribution actually measured by a temperature sensor on the

chip. With this, the precision in heat release control is further improved.

The processor according to the embodiments may be mounted on a substrate together with other components such as a memory so as to form a processor system. An information processing apparatus may be formed in which such a processor system is installed. A personal computer and a variety of portable equipment are examples of such information processing apparatuses.

INDUSTRIAL APPLICABILITY

The present invention is applicable to the field of heat control in a processor.